Duplexing structure of switching system processor and method for maintaining memory coherency

Patent Number:	□ US6594778
Publication date:	2003-07-15
Inventor(s):	KIM IN-CHUL (KR)
Applicant(s):	LG INF & COMM LTD (KR)
Requested Patent:	CN1285694
Application Number:	US20000617495 20000717
Priority Number(s):	KR19990035264 19990824
IPC Classification:	G06F7/02
EC Classification:	
Equivalents:	KR2001019058

Abstract

A duplexing structure of a switching system processor and a method for maintaining a memory coherency are provided in which both data of an active memory and of a standby memory are concurrently read. The concurrently read data can be compared using hardware. In the duplexing structure, an address corresponding to the concurrent reading operation is set and a path is provided therefore. Accordingly, when an address of the active memory and the standby memory are compared, they are simultaneously accessed and compared for judgement at one time. Thus, a time required for the operation is reduced. In addition, since the comparison result can be recognized according to the kinds of the termination that is applied by hardware, without comparing the two data relying on a series of program operations on the basis of software using the CPU, the load of the CPU required with respect to memory coherency verification can be reduced

Data supplied from the **esp@cenet** database - l2

Description

BACKGROUND OF THE INVENTION

[0002] 1. Field of the Invention

[0003] The present invention relates to a switching system, and more particularly, a duplexing processor in a switching system and method thereof.

[0004] 2. Background of the Related Art

[0005] In general, a switching system includes a plurality of processors, which are required to process data on a real time basis, and are managed by a duplexing channel to prevent an interruption in processing of data even when a malfunction occurs. A duplexing channel is generally formed through a back plane between two processor boards having the same construction. While one processor is operating in an active mode, the other processor is operating in a standby mode. When a disturbance occurs at the active mode processor, the standby mode processor is switched to an active mode to successively perform the data processing.

[0006] As the processor is switched oil duplexing, the memories of the two processors should be identical to each other in order for the CPU of the standby processor to successively perform its normal operation. For this purpose, when the CPU of the active processor performs a data writing operation on its memory, it performs a concurrent writing operation when the data is one to be concurrently written into the memory of the standby processor. Accordingly, the same data is written into the same address of the memory of the active processor and into the memory of the standby processor. In order to identify whether the concurrent writing was successfully performed, the CPU of the active processor periodically compares the two memories of the two processor, thereby maintaining a coherency of the memories of the active processor and the standby processor.

[0007] The operation of the background art duplexing processor will now be described with reference to FIG. 1. The duplexing processor of the switching system consists of an active processor 10 and a standby processor 20 that are connected to each other through a duplexing bus. The active processor 10 and the standby processor 20 respectively include a CPU 11 and 21, a memory 12 and 22, a concurrent write logic unit 13 and 23, and a standby read logic unit 14 and 24.

[0008] The CPU 11 of the active processor 10 performs three kinds of operations: (1) a self-reading and writing operation for reading or writing only its own memory (active memory 12); (2) a concurrent writing operation for concurrently writing the same data also on the memory (standby memory 22) of the standby processor when writing the data on the active memory 12; and (3) a standby reading operation for reading a data of a corresponding arbitrary address of the standby memory 22 to compare it with the data written in the arbitrary address of the active memory 12 to check whether the concurrent writing operation was correctly performed.

[0009] The three operations are divided by an agreement between a software and a hardware. For example, regarding '68000/68020/68030' type CPUs in the line of 68XXX processors, the three operations are performed according to classification of an address field value, and a logic of the hardware is driven according to the address field value. While the three operations are performed by a UPA0/UPA1 (UPA: User Page Attributes) of the CPU with respect to '68040/68060'.

[0010] For example, on the assumption of 128 Mbyte memory, the operation of the CPU are as shown Table 1 as follows:

TABLE 1
68000/68020/6803068040/68060
address field)(UPA bit)
(1) Self-reading and0x000000000x07FFFFFFUPA1:UPA0 = '00'
writing
(2) Concurrent writing0xC00000000xC7FFFFFFUPA1:UPA0 = '01'
(3) Standby reading0x400000000x47FFFFFFUPA1:UPA0 = '10'
(4) ReservedUndefinedUPA1:UPA0 = '11'

[0011] In this manner, when the operation state of the CPU 11 is represented according to the setting of the address field value or the UPA bit value previously determined on the basis of software, the hardware drives logic that is suitable for the operation.

[0012] When an address field value or a UPA bit value corresponding to the '(2) concurrent writing' operation of Table 1 is applied from the CPU 11, the concurrent write logic unit 13 accesses the memory 22 of the standby processor through the duplexing bus and writes the same data into the two memories 12 and 22. When an address field value or a UPA bit value corresponding to the '(3) standby reading' operation of Table 1 is applied from the CPU 11, the standby read logic unit 14 accesses the memory 22 of the standby processor through the duplexing bus and reads the data of the memory 22 to transfer it to the CPU 11. The three operations of the background art duplexing processor will now be described.

[0013] 1. Self Reading and Writing Operation

[0014] This operation is performed when the data written in the active memory 12 is desired to be read by the CPU 11, and when the data is desired to be written only in the active memory 12. That is, the data written into the memory 12 is not written into the standby memory 22 because the data wouldn't affect the operation of the CPU 22.

[0015] First, the CPU 11 sets an address field value or a UPA bit value corresponding to the '(1) Self reading and writing' operation of Table 1 and selects an address desired to be read or written. The units 13 and 14 corresponding to the duplexing logic are not driven, and only the corresponding active memory 12 is accessed. Thus, the data of the corresponding address is read or written.

[0016] 2. Concurrent Writing Operation

[0017] This operation is performed when the data is desired to be concurrently written into the active memory 12 and the standby memory 22 by the CPU 11. First, the CPU 11 sets an address field value or a UPA bit value corresponding to the '(2) concurrent writing' operation of Table 1 and outputs a data to be written and its address. Thereafter, the data is written into the corresponding address of the active memory 12, and simultaneously, the concurrent write logic unit 13 is driven.

[0018] The concurrent write drive unit 13 accesses the memory 22 of the standby processor and transfers the same data as the data transferred to the active memory 12 through the duplexing bus. Accordingly, the same data as that written in the active memory 12 is written into the corresponding address of the standby memory 22.

[0019] 3. Memory Coherency Maintaining Operation

[0020] This operation is performed to periodically identify whether the same data was written in the active memory 12 and in the standby memory 22 by means of the CPU 11 of the active processor 10. Memory coherency enables the CPU 21 of the standby processor 20 to be able to continuously perform the operation of the previous active processor in case of the duplexing switch.

[0021] First, the active memory 12 is accessed and a data written in the corresponding address of the memory 12 is applied to the CPU 11. Thereafter, when the CPU 11 selects the address field value or the UPA bit value corresponding to the '(3) standby read' operation of Table 1 and the same address as above, the standby read logic unit 14 is driven.

[0022] The standby read logic unit 14 accesses the memory 22 of the standby processor, reads a data of a corresponding address of the memory 22 and transfers it to the CPU 11 through the duplexing bus. Thereafter, the CPU 11 compares the data as read from the active memory 12 and the data as read from the standby memory 22. If the two data are identical, the CPU 11 repeatedly performs the above process to thereby compare the data of the next address.

[0023] Meanwhile, upon such comparison, if the two data of the memories 12 and 22 are not identical, the CPU 11 sets a field value of the discordant address or a UPA bit value corresponding to the '(2) concurrent writting' operation of Table 1. Accordingly, the concurrent write logic unit 13 is driven so as to transfer a data of a predetermined address of the active memory 12 to the standby memory 22 through the duplexing bus. In this manner, the same data as the data written in the active memory 12 is written into the corresponding address where the discordance occurred of the standby memory 22.

[0024] FIGS. 2A and 2B show views of timing of an example of the memory coherency maintaining operation of the duplexing processor. FIG. 2A shows the case where the two memories are identical, and FIG. 2B shows the case where the two memories are not identical.

[0025] Referring to FIGS. 2A and 2B, in the case where the CPU 11 compares the data of a specific address, i.e., '0x1000', it sets the address as '0x1000' to represent a self-reading operation and accesses the active memory 12 to read a corresponding address value. Thereafter, the CPU 11 sets the address as '0x40001000' so as to represent a standby reading operation, and drives the standby read logic unit 14 to receive the corresponding address value of the standby memory 22.

[0026] The CPU 11 compares the data read from the two memory 12 and 22. As shown in FIG. 2A, if the two data as read are equal(i.e., '0x0123'), it sets the next address to be compared (i.e., '0x1004'). The CPU 11 then repeatedly performs the same process described above.

[0027] Meanwhile, as shown in FIG. 2B, if the two data as read are not equal (i.e., '0x0123' and '0x012F'), the CPU 11 again reads the data '0x0123' of the corresponding address '0x1000' of the active memory 12 and then drives the concurrent write logic unit 13 to concurrently writes the data '0x0123' into the active memory 12 and the standby memory 22.

[0028] Thereafter, the CPU 11 performs comparing operation for the next address (i.e., '0x1004') using the same process as described above. Thus, the process for maintaining the memory coherency in the background art duplexing processor is performed by reading the data of the active memory, reading the data of the standby memory, comparing the two data, and performing the concurrently writing operation according to the comparison result.

[0029] As described above, the background art duplexing processor has various disadvantages. The CPU of the active processor performs each step by contiguously changing the address values needed to perform each operation and compares the data on the basis of software, which results in a lengthy time for the operation and a load of the CPU is increased. Thus, because of the verification time limit and the restraint of the load of the CPU, it is hard to perform such a verification function while the processor is actually in service (i.e., processing of a call of actual subscribers, or billing). Accordingly, the reliability of the processor is reduced.

[0030] The above references are incorporated by reference herein where appropriate for appropriate teachings of additional or alternative details, features and/or technical background.

SUMMARY OF THE INVENTION

[0031] An object of the invention is to solve at least the above problems and/or disadvantages and to provide at least the advantages described hereinafter.

[0032] Another object of the present invention is to provide a switching system processor and method that substantially obviates one or more of the problems caused by disadvantages in the background art.

[0033] Another object of the present invention is to provide a duplexing processor operated in a standby and active mode that maintains coherent memory and method thereof.

[0034] Another object of the present invention is to provide a duplexing structure of a switching system processor and method in which both data of an active memory and of a standby memory are concurrently read and the data as read are compared on the basis of hardware for maintaining a memory coherency.

[0035] To achieve these objects and other advantages in a whole or in part and in accordance with the purpose of the present invention, as embodied and broadly described herein, there is provided a switching system having a duplexing channel between two processors each adapted to operate in an active mode and a standby mode, each processor that includes a memory, a CPU that generates a control signal for concurrently reading a selected memory address to be compared, a concurrent read logic unit that concurrently accesses memories of an active processor and a standby processor responsive to the control signal of the CPU and reads data from the selected addresses in the memories, a comparator that compares the two data from the selected address of the memories of the two processors, and a concurrent read termination generator that generates a prescribed termination signal according to a comparison result by the comparator.

[0036] To further achieve these objects and other advantages in a whole or in part and in accordance with the purpose of the present invention, as embodied and broadly described herein, there is provided a method for maintaining a memory coherency of a switching system processor having a duplexing channel between two processors that each mutually operate in an active mode and a standby mode, respectively, that includes generating a prescribed control signal to perform a concurrent reading operation for an arbitrary memory address, concurrently accessing memories of the active and standby processors, respectively, to read data from the arbitrary address of both memories responsive to the control signal, comparing the data read from the two processor memories, and generating a termination signal according to a comparison result.

[0037] To further achieve these objects and other advantages in a whole or in part and in accordance with the purpose of the present invention, as embodied and broadly described herein, there is provided a switching system having a data bus between a plurality of processors each adapted to operate in an active mode and a standby mode, wherein at least one processor is in the active mode mid remaining processors are in the standby mode, each processor that includes a memory, a CPU that generates a control signal for concurrently reading a selected memory address to be compared, a concurrent read logic unit that concurrently accesses memories of the at least one active processor and the standby processors responsive to the control signal of the CPU and reads data from the selected addresses in the memories, a comparator that compares the data from the selected address of the memories of the processors, and a concurrent read termination generator that generates a prescribed termination signal according to a comparison result by the comparator.

[0038] Additional advantages, objects, and features of the invention will be set forth in part in the description which follows and in part will become apparent to those having ordinary skill in the art upon examination of the following or may be learned from practice of the invention. The objects and advantages of the invention may be realized and attained as particularly pointed out in the appended claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0039] The invention will be described in detail with reference to the following drawings in which like reference numerals refer to like elements wherein:

[0040] FIG. 1 is a schematic block diagram of a background art switching system processor as implemented by duplexing;

[0041] FIGS. 2A and 2B are timing diagrams showing an example of a process of maintaining a memory coherency of the background art duplexing processor of FIG. 1;

[0042] FIG. 3 is a diagram showing a preferred embodiment of a switching system processor in accordance with the present invenition; and

[0043] FIGS. 4A and 4B are timing diagrams showing a preferred embodiment of a process of maintaining a memory coherency of a duplexing processor in accordance with the present invention

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

[0044] FIG. 3 is a schematic block diagram showing internal construction of a first preferred embodiment of a switching system processor in accordance with the present invention. The first preferred embodiment of a switching system processor is implemented on a duplexer processor, however, the present invention is not intended to be so limited. As shown in FIG. 3, an active processor

30 and a standby processor 40 of a duplexing processor in accordance with the first preferred embodiment respectively include CPUs 31 and 41, memories 32 and 42, concurrent write logic units 33 and 43, standby read logic units 34 and 44, memory compare units 35 and 45, and concurrent read termination generation units 36 and 46.

[0045] An address field value or a UPA bit value for performing a concurrent reading is pre-defined for the.CPUs 31 and 41 by preferably using values that were previously not defined. As described above, the address field value or the UPA bit value for performing a self reading and writing operation, a concurrent writing operation and a standby reading operation are already defined in the background art as shown in Table 1.

[0046] For example, an address field value of '0x20000000' may be used for "concurrent reading" in the case of 68000/68020/68030 type CPUs in the line of 68XXX processors, and a UPA bit value of '11' may be used for the case of '68040/68060'. As shown in Table 1, the address field values 0x200000000x27FFFFFF are not yet defined

[0047] When a selected one of the CPUs 31 and 41 sets an address value corresponding to the concurrent reading, the memory compare units 35 and 45 simultaneously receives the data for the address from its own memory and the memories 32 and 42 at the other side (i.e., processor) and compares the two data to judge whether they are identical to each other on the basis of hardware. In this respect, the reading of the memory at the other side is preferably performed by the standby read logic units 34 and 44 without the necessity of any additional logic. Further, in this case, the CPU 31 or 41 is selected by being in the active mode.

[0048] The concurrent read termination generation units 36 and 46 preferably checks whether a normal (successful) operation was performed according to the comparison result by means of the memory compare units 35 and 45, and transfers it to the CPUs 32 and 42.

[0049] Operations of the memories 32 and 42, the concurrent write logic units 33 and 43, and the standby read logic units 34 and 44 are the same as those in the background art. Accordingly, a detailed description is omitted.

[0050] A second preferred embodiment in accordance with the present invention of a process for maintaining memory coherency of a switching system processor will now be described. The second preferred embodiment of the process for maintaining memory coherency can be used, for example, in the first preferred embodiment of a duplexing processor. However, the present invention is not intended to be so limited. The second preferred embodiment of the process for maintaining memory coherency implemented by duplexing will be described wvith respect to the duplexing processor of FIG. 3.

[0051] First, the CPU 31 sets an address field value or a UPA bit value corresponding to the concurrent reading as pre-defined and a memory address desirous of coherency validation. Thereafter, the data written in the corresponding address of the active memory 32 is applied to a terminal of the memory compare unit 35, and simultaneously, the standby read logic unit 34 is driven.

[0052] The standby read logic unit 34 accesses the memory 42 of the standby processor, reads the data of the corresponding address of the memory 42 and transfers it to the memory compare unit 35 through the duplexing bus. The memory compare unit 35 compares the data as read from the active memory 32 and the data as read from the standby memory 42, and if the two data are identical to each other, the memory compare unit 35 generates a predetermined match signal.

[0053] Accordingly, the concurrent read termination generation unit 36 applies a termination signal to the CPU 31 indicating the operation was normally performed to the CPU 31. Then, the CPU 31 repeatedly performs the process as described above to compare the next address.

[0054] However, if upon comparison the data from the two memories 32 and 42 compared by memory compare unit 35 are not identical to each other, the memory compare unit 35 generates a predetermined mismatch signal. Accordingly, the concurrent read termination generation unit 36 applies a signal to the CPU 31 indicating an abnormal operation was performed to the CPU 31. In this case, the CPU 31 sets an address field value corresponding to the self reading and writing operation, reads a data written in the active memory 32, and again resets an address field value corresponding to the concurrent writing operation. Then, the CPU 31 performs the concurrent writing operation into the active memory 32 and the standby memory 42.

[0055] In this manner, the coherency for the corresponding addresses of the two memories 32 and 42 can be maintained without a separate access in time of the processors in the switching system. The CPU 31 preferably repeatedly performs the same processes as described above for the next address to be compared.

[0056] FIGS. 4A and 4B are views of timing showing a process of maintaining a memory coherency of the duplexing processor in accordance with the second preferred embodiment of the present invention. FIG. 4A shows the case that the data of the addresses desired to be compared of the two memories 32 and 42 are identical, and FIG. 4B shows the case that the data of the addresses desired to be compared of the two memories 32 and 42 are not identical. As shown in FIGS. 4A and 4B, an address field value representing the concurrent reading is defined as '0x2000000-0x27FFFFFF'.

[0057] As shown in FIG. 4A to compare data of each address of '0x1000', the CPU 31 preferably generates an address of '0x2000100' representing the concurrent reading operation for the '0x1000' address. The data of the active memory 32 corresponding to the '0x1000' address is transferred to the memory compare unit 35. Simultaneously, the standby read logic unit 34 is driven so that the data of '0x1000' address of the memory 42 of the standby processor 40 is transferred to the memory compare unit 35 through the duplexing bus.

[0058] The memory compare unit 35 compares the data as read from the memory 32 and from the memory 42. As shown in FIG. 4A, when the two data as read are the same (e.g., '0x0123'), the memory compare unit 35 does not generate an error or a MISMATCH* signal. Thus, the concurrent read termination generation unit 36 transfers a termination signal, that is for example, a DTACK* signal or TA* signal, informing the CPU 31 that the normal bus operation was completed.

[0059] Accordingly, the CPU 31 identifies that the '0x1000' addresses of the two memories 32 and 42 are identical to each other, sets an address value as '0x20001004' for comparing the next address (e.g., a '0x1004' address) to be compared and repeatedly performs the above described processes.

[0060] As shown in FIG. 4B, when the two data respectively transferred from the '0x1004' addresses of the two memories 32 and 42 are not identical in the concurrent reading operation (e.g., '0x4567' and '0x456F'), the memory compare unit 35 generates the error or the MISMATCH* signal. The concurrent read termination generation unit 36 preferably transfers, for example a BERR* or a TEA* signal to the CPU 31, informing that an abnormal bus operation was performed.

[0061] Accordingly, the CPU 31 identifies that the '0x1004' address of the two memories 32 and 42 are not identical to each other and performs an operation to make them identical. For this purpose, the CPU 31 preferably sets an address as '0x00001004' corresponding to the self-reading operation for '0x1004' and reads the data of the '0x1004' address of the active memory 32. Thereafter, the CPU 31 sets the address as '0x00001004' to perform the concurrent writing for the address. Then, the concurrent write logic unit 33 is driven and the same data as that of the active memory 32 are simultaneously written in the standby memory 42 to maintain coherency for the address. Again, the CPU 31 can repeatedly perform the above described processes for the next address to be compared.

[0062] As described above, preferred embodiments of a switching system processor (e.g., duplex) and method of memory coherency thereof according to the present invention have various advantages. According to the preferred embodiment of a duplexing structure of a switching system processor and a method for maintaining memory coherency, an address corresponding to the concurrent reading operation is set and a path is provided therefore. Accordingly, when the active memory and the standby memory are compared, the memories are simultaneously accessed and compared for judgement at one time, unlike that of the background art where the two memories are separately accessed. Thus, a time required for the operation is much reduced. Further, the result can be recognized according to the kinds of the termination that is applied by hardware without comparing the two data relying on a series of program operations on the basis of software using the CPU so that the load of the CPU required with respect to memory coherency verification can be reduced. Accordingly, the concurrent reading operation of the preferred embodiments can be efficiently performed during actual service by the CPU.

[0063] The foregoing embodiments and advantages are merely exemplary and are not to be construed as limiting the present invention. The present teaching can be readily applied to other types of apparatuses. The description of the present invention is intended to be illustrative, and not to limit the scope of the claims. Many alternatives, modifications, and variations will be apparent to those skilled in the art. In the claims, means-plus-function clauses are intended to cover the structures described herein as performing the recited function and not only structural equivalents but-also equivalent structures.

Data supplied from the **esp@cenet** database - I2

Claims

What is claimed is:

[0064] 1. A switching system having a duplexing channel between two processors each adapted to operate in an active mode and a standby mode, each processor comprising:a memory; a CPU that generates a control signal for concurrently reading a selected memory address to be compared; a concurrent read logic unit that concurrently accesses memories of an active processor and a standby processor responsive to the control signal of the CPU and reads data from the selected addresses in the memory of the active processor and the memory of the standby processor; a comparator that compares the two data from the selected address of the memories of the active and standby processors; and a concurrent read termination generator that generates a prescribed termination signal according to a comparison result by the comparator.

[0065] 2. The system of claim 1, wherein a field value of the address for performing the concurrent reading operation is defined in the CPU.

[0066] 3. The system of claim 2, wherein additional field values of the address are respectively 0000000007FFFFFF for a self read/write operation, C00000007FFFFFF for a concurrent write operation and 4000000047FFFFFF for a standby read operation.

[0067] 4. The system of claim 1, wherein the concurrent read termination generator of the active processor applies one of a normal termination signal to the CPU of the active processor when the comparison result indicates the two data are equal and a bus error termination signal otherwise.

[0068] 5. The system of claim 1, wherein the CPU of the active processor controls the concurrent reading process for a next address to be compared responsive to a normal termination signal from the concurrent read termination generator for the selected address.

[0069] 6. The system of claim 1, wherein the CPU reads the data from the selected address of the memory of the active processor and then concurrently rewrites the data from the selected address of the memory of the active processor in the selected address of the memories of the active and standby processors responsive to an abnormal termination signal from the concurrent read termination generator.

[0070] 7. The system of claim 1, further comprising a concurrent write logic unit that concurrently writes data in the memories of the active and the standby processors under control of the CPU.

[0071] 8. The system of claim 1, wherein each processor comprises a processor board.

[0072] 9. A method for maintaining a memory coherency of a switching system processor having a duplexing channel between two processors that each mutually operate in an active mode and a standby mode, respectively, comprising:generating a prescribed control signal to perform a concurrent reading operation for an arbitrary memory address; concurrently accessing memories of the active and standby processors, respectively, to read data from the arbitrary address of both memories responsive to the control signal; comparing the data read from the two processor memories; and generating a termination signal according to a comparison result.

[0073] 10. The method according to claim 9, wherein an address field value for performing the concurrent reading operation is defined in a CPU of the processors to generate the prescribed control signal.

[0074] 11. The method of claim 10, wherein address field values are defined for a self read/write operation, a concurrent write operation and a standby read operation in the CPU.

[0075] 12. The method according to claim 9, wherein the generating the termination signal comprises:generating a normal terminal signal when the two data are equal according to the comparison result and an abnormal termination signal when the two data are not equal; and applying the termination signal to the active CPU.

[0076] 13. The method according to claim 12, further comprising performing the concurrent read process for a next address relative to the arbitrary address.

[0077] 14. The method of according to claim 13, wherein when the normal termination signal is applied, the generating a prescribed signal step through generating a termination signal step comprise the performing the concurrent read process for the next address.

[0078] 15. The system according to claim 13, wherein when the abnormal termination signal is applied, the data is read from the arbitrary address of the memory of the active processor and then concurrently rewritten in the arbitrary addresses of the memories of the active and the standby processors before the performing the concurrent read operation for the next address.

[0079] 16. A switching system having a data bus between a plurality of processors each adapted to operate in an active mode and a standby mode, wherein at least one processor is in the active mode and remaining processors are in the standby mode, each processor comprising:a memory; a CPU that generates a control signal for concurrently reading a selected memory address to be compared; a concurrent read logic unit that concurrently accesses memories of the at least one active mode processor and the standby mode processors responsive to the control signal of the CPU of the at least one active mode processor and reads data from the selected addresses in the memories; a comparator that compares the data from the selected address of the memories of the processors; and a concurrent read termination generator that generates a prescribed termination signal according to a comparison result by the comparator.

[0080] 17. The switching system of claim 16, wherein a field value of the address for performing the concurrent reading operation is defined in the CPU.

[0081] 18. The switching system of claim 16, wherein the concurrent read termination generator of the active mode processor applies one of a normal termination signal to the CPU of the active mode processor when the comparison result indicates the two data are equal and a bus error termination signal otherwise, and wherein the CPU of the active mode processor controls the concurrent reading process for a next address to be compared responsive to a normal termination signal from the concurrent read termination generator for the selected address.

[0082] 19. The switching system of claim 16, wherein the CPU reads the data from the selected address of the memory of the active mode processor and then concurrently rewrites the data from the selected address of the memory of the active mode processor in the selected address of the memories of the active and standby mode processors responsive to an abnormal termination signal from the concurrent read termination generator.

[0083] 20. The switching system of claim 16, further comprising a concurrent write logic unit that concurrently writes data in the memories of the active and the standby mode processors under control of the CPU.

---

[19]中华人民共和国国家知识产权局

[51]Int. Cl⁷
H04Q 3/00

[12]发明专利申请公开说明书

[21]申请号 00121524.8

[43]公开日 2001年2月28日

[11]公开号 CN 1285694A

[22]申请日 2000.8.9 [21]申请号 00121524.8
[30]优先权
[32]1999.8.24 [33]KR [31]35264/1999
[71]申请人 LG 情报通信株式会社
地址 韩国汉城市
[72]发明人 金寅澈

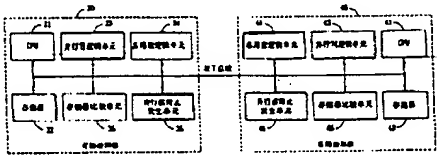
[74]专利代理机构 中原信达知识产权代理有限责任公司
代理人 余朦 李辉

权利要求书3页 说明书11页 附图页数5页

[54]发明名称 保持存储器相干性的交换系统处理器的双工结构和方法

[57]摘要

一种用于保持存储器相干性的交换系统处理器的双工结构和方法，其中并行地读取有效存储器和备用存储器的数据。利用硬件比较并行读取的数据。在双工结构中，设置相应于并行读取操作的地址并因此提供一条路径。当将有效存储器和备用存储器的地址进行比较时，对它们同时访问和比较以同时进行判断。因此，减少了操作所需的时间。按照由硬件加载的终止类型可以识别该结果而不必依靠使用CPU根据软件的一系列程序操作来比较这两个数据。



ISSN 1008-4274

知识产权出版社出版

存储器的双工处理器及其方法。

本发明的另一目的是提供交换系统处理器的一种双工结构和方法，其中，为了保持存储器相干性并行地读取有效存储器和备用存储器的数据并且依靠硬件比较读取的数据。

为了全部或部分地达到这些目的和其他优点并按照本发明的目的，正如在此处具体表达并广泛地描述的一样，提供一种具有在各自适合于以有效模式和备用模式操作的两个处理器之间的双工通道的交换系统，每个处理器包括：存储器；CPU，其产生用于并行地读取将要比较的选定的存储器地址的控制信号；并行读逻辑单元，其响应于CPU的控制信号并行地访问有效处理器和备用处理器的存储器并从存储器中的选定地址读取数据；比较器，其将来自两个处理器的存储器的选定地址的两个数据进行比较；以及并行读终止发生器，其根据比较器的比较结果产生规定的终止信号。

为了更进一步地全部或部分地达到这些目的和其他优点并按照本发明的目的，正如在此处具体表达并广泛地描述的，提供一种用于保持交换系统处理器的存储器相干性的方法，该交换系统处理器具有在各自相互以有效模式和备用模式操作的两个处理器之间的双工通道，该方法包括：产生一个规定的控制信号以便执行任意的存储器地址的并行读操作，并行地分别访问有效处理器和备用处理器的存储器，以便响应于该控制信号从两个存储器的任意地址读取数据，比较从两个处理器存储器读取的数据，以及根据比较结果产生终止信号。

为了更进一步地全部或部分地达到这些目的和其他优点并按照本发明的目的，正如在此处具体表达并广泛地描述的，提供一种具有在各自适合于以有效模式和备用模式操作的多个处理器之间的数据总线的交换系统，其中至少一个处理器在有效模式下而其余处理器在备用模式下，每个处理器包括：存储器；CPU，其产生用于并行地读取将

要比较的选定的存储器地址的控制信号；并行读逻辑单元，其响应于CPU的控制信号并行地访问该至少一个有效处理器和其它备用处理器的存储器并从存储器中的选定地址读取数据；比较器，其将来自处理器的存储器的选定地址的数据进行比较；以及并行读终止发生器，其根据比较器的比较结果产生规定的终止信号。

本发明的另外的优点、目的和特点将在随后的叙述中部分地阐明并将部分地在对下面内容的审看中由那些本领域的普通技术人员理解或者可以从本发明的实践中掌握。可以如在附加的权利要求书中特别指出的一样实现并获得本发明的目的和优点。

下面，将参考下列附图详细描述本发明，在附图中相同的参考数字是指其中相同的单元：

图 1 是通过双工实现的背景技术的交换系统处理器的示意方框图；

图2A和2B是表示图 1 的背景技术的双工处理器的存储器相干性的保持过程的例子的时序图；

图 3是表示按照本发明的交换系统处理器的最佳实施例的图；

图4A和4B是表示按照本发明的双工处理器的存储器相干性的保持过程的最佳实施例的时序图。

图 3是表示按照本发明的交换系统处理器的第一个最佳实施例的内部构造的示意方框图。在一双工处理器上实现交换系统处理器的第一个最佳实施例，但是，本发明并不受限于此。如图3所示，按照本发明的第一个最佳实施例的双工处理器的有效处理器30和备用处理器40分别包括：CPU31和41，存储器32和42，并行写逻辑单元33和43，备用读逻辑单元34和44，存储器比较单元35和45以及并行读终止发生单元36和46。

最好通过使用没有预先确定的值来为CPU 31和41预定用于执行

并行读的地址字段值或UPA比特值。如上所述，在如表1所示的背景技术中已经确定了用于执行自读和写操作、并行写操作和备用读操作的地址字段值或UPA比特值。

例如，在 68XXX处理器的系列中的 68000/68020/68030型号CPU的情况下，' 0x20000000 '的地址字段值可以用来 "并行读取"，而‘11’的UPA比特值可以用于 ' 68040/68060 '的情况。如表1所示，地址字段值0x20000000~0x27FFFFFF还没有确定。

当CPU 31和41中的选定的那个相应于并行读取而设置地址值时，存储器比较单元35和45从它们自己的存储器和另一侧(即，处理器)的存储器32和42处同时地接收地址的数据而并将两个数据进行比较以便根据硬件来判断它们是否彼此相同。在这方面，最好通过备用读逻辑单元34和44而无需任何附加的逻辑来执行在另一侧的存储器的读取。在这种情况下，通过处于有效模式来选择CPU 31或41。

并行读终止发生单元36和46最好根据存储器比较单元35和45的比较结果检查是否执行了一个正常的 (成功的)操作，并传输到CPU 32和42。

存储器32和42、并行写逻辑单元33和43以及备用读逻辑单元34和44的操作与背景技术中的操作相同。因此，省略了其详细叙述。

现在将描述按照本发明的第二个最佳实施例的用于保持交换系统处理器的存储器相干性的过程。例如，用于保持存储器相干性的过程的第二个最佳实施例可用于双工处理器的第一个最佳实施例中。可是，本发明不意指受限于此。下面，将相对于图 3的双工处理器描述通过双工实现的用于保持存储器相干性的过程的第二个最佳实施例。

首先，相应于预确定的并行读取和期望的相干性检验的存储器地

址，CPU 31设置地址字段值或UPA比特值。其后，写入有效存储器32的相应地址的数据被加载到存储器比较单元35的端子，并且同时驱动备用读逻辑单元34。

备用读逻辑单元34访问备用处理器的存储器42，通过双工总线读取存储器42的相应地址的数据并将其传输到存储器比较单元35。存储器比较单元35把从有效存储器32读取的数据和从备用存储器42读取的数据进行比较，如果两个数据彼此相同，则存储器比较单元35产生一个预先确定的匹配信号。

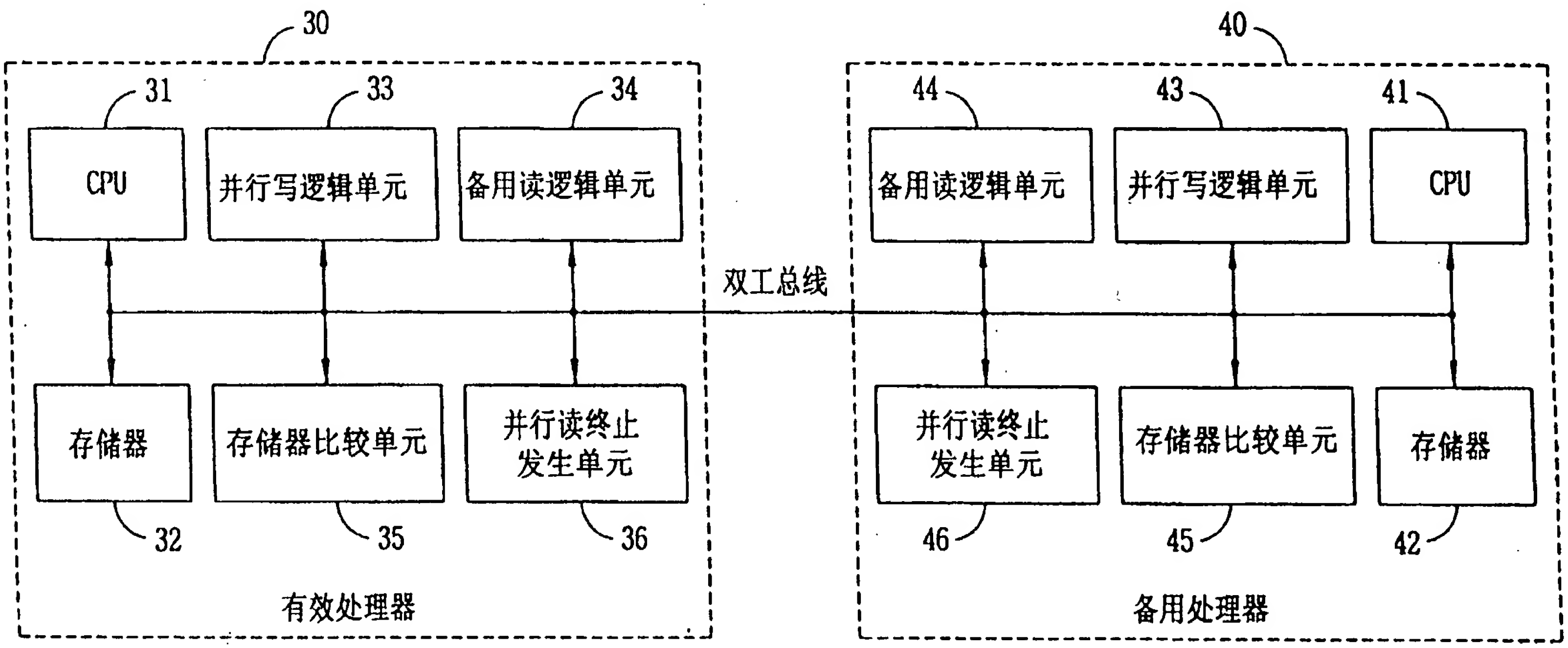
因此，并行读终止发生单元36将终止信号加载到CPU 31，向CPU 31指示操作正常执行。然后，CPU 31如上所述地执行该过程以便比较下一个地址。

可是，如果存储器比较单元35比较的来自两个存储器32和42的数据彼此不相同，则存储器比较单元35产生一个预先确定的失配信号。从而，并行读终止发生单元36将信号加载到CPU 31，向CPU 31指示执行了一个不正常的操作。在这种情况下，CPU 31相应于自读和写操作，设置地址字段值，读取写入有效存储器32的数据，并相应于该并行写操作再一次设置地址字段值。然后，CPU 31执行向有效存储器32和备用存储器42的并行写操作。

以这种方式，可以保持两个存储器32和42的相应地址的相干性而不必在交换系统中的处理器的时间上的分别访问。然后，为了比较下一个地址，CPU 31最好反复执行如上所述的相同过程。

图4A和4B是表示按照本发明第二个最佳实施例的双工处理器的存储器相干性的保持过程的时序图。图4A表示两个存储器32和42期望比较的地址的数据完全相同的情况，而图4B表示两个存储器32和42期望比较的地址的数据不相同的情况。如图4A和4B所示，表示并行读取

图3


30
31
CPU
33
并行写逻辑单元
34
备用读逻辑单元
存储器
32
存储器比较单元
35
并行读终止
发生单元
36
有效处理器
双工总线
40
44
备用读逻辑单元
43
并行写逻辑单元
41
CPU
并行读终止
发生单元
46
存储器比较单元
45
存储器
42
备用处理器